list item properties:

are ordered, are mutable, can be different data types

write the membership operator

element in my\_list #returns True or False

Cloning a list (B = A will not work!) B = A[:] / my\_list.copy()

given: myList = [‘a’, ‘b’, ‘c’, ‘d’]

What do each of the following operations do (retain changes)?

myList[1:2] = [‘x’, ‘y’] a, x, y, c, d (fits all elements in)

myList[1:4] = [‘s’] a, s, d (still removes the 1:4 slice)

given: myList = [‘a’, ‘b’, ‘c’], write code for the following ops

Add ‘e’ after ‘b’: myList.insert(2, ‘e’)

Add ‘k’ at the end: myList.append(‘k’)

Change ‘a’ to ‘t’ myList[0] = ‘t’

Results of x, \*y, z = myList: x = ‘t’, y = [‘b’, ‘e’, ‘c’], z = ‘k’

Add elements of other iter: myList.extend(otherList)

Remove ‘c’ from list: myList.remove(‘c’)

Remove ‘s’ from list: Will cause error if ‘s’ doesn’t exist in list!

Remove last item: myList.pop() / del myList[-1]

Remove first item: myList.pop(0) / del myList[0]

Turn list into set: set(myList)

(to remove dupes)

Find length of list: len(myList)

Sum of all elements: sum(myList)

Clear entire list: myList.clear()

Loop over a list backwards: item in ( reversed(myList) / myList[::-1] )

Actually reverse the list: myList.reverse()

Print every other element starting at index 1: for item in my\_list[1::2]:

print(item)

Find first index the letter ‘e’ appears: myList.index(‘e’)

Sort list in reverse: myList.sort(reverse=True)